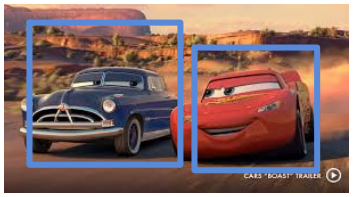
Object Localization

Classification with localization

Image classification:

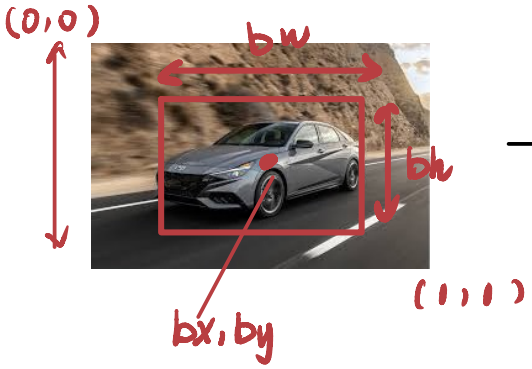
"Car"



Detection:

- multiple objects
- (Sometimes) multiple categories.

classification with localization



→ ConvNet →

→ Softmax (4)

1. car
2. pedestrain
3. motorcycle
4. none

→ bx, by, bh, bw

bounding box.

Defining the target label y

$$y = \begin{bmatrix} P_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$$

← is there an object?

$$\begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

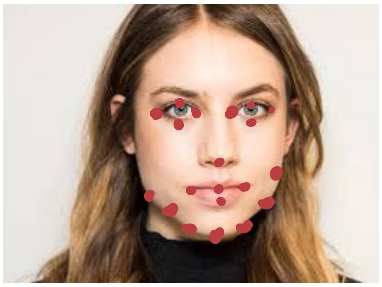
No object

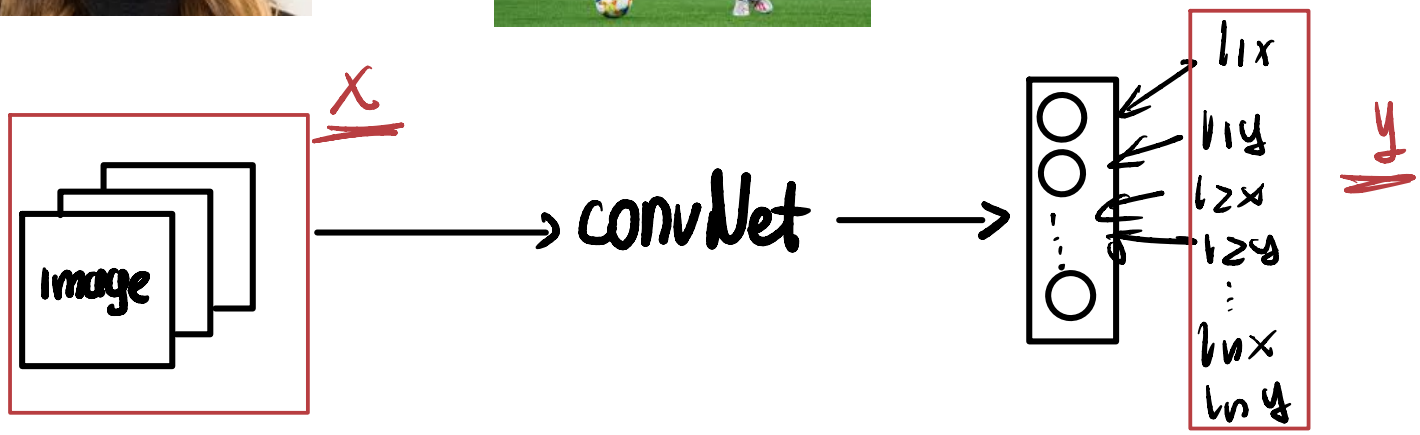
"Don't care"

$$\begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix}$$

$$L(\hat{y}, y) = \begin{cases} (\hat{y}_1 - y_1)^2 + (\hat{y}_2 - y_2)^2 + \cdots (\hat{y}_n - y_n)^2 & \text{if } y_1 = 1 \\ (\hat{y}_1 - y_1)^2 & \text{if } y_1 = 0 \end{cases}$$

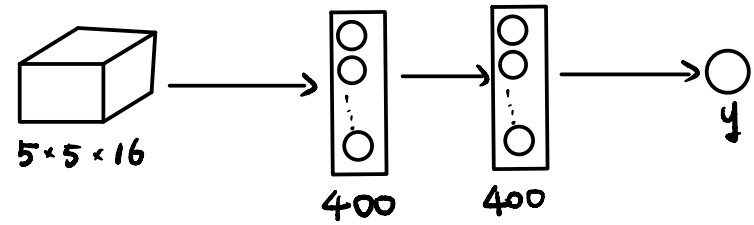
Landmark Detection

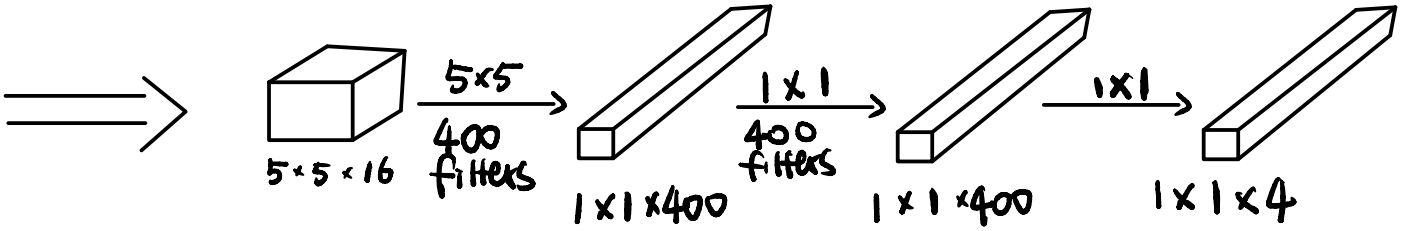




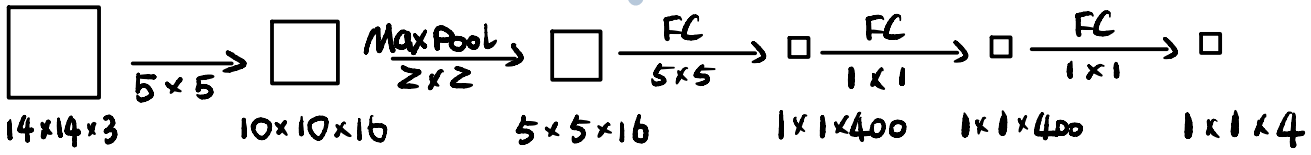
object Detection

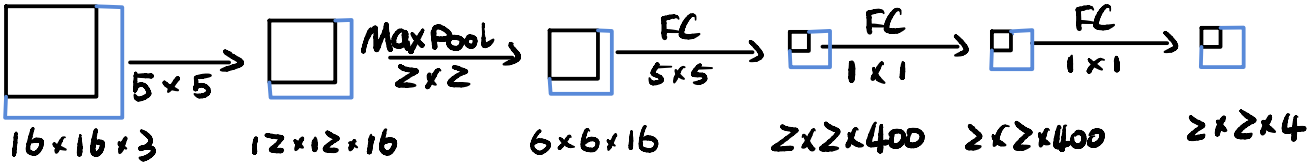
Tuning FC layer into Convolutional layers :



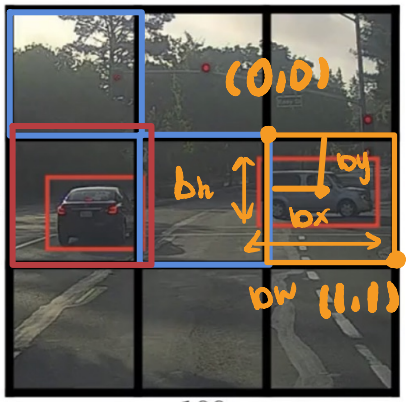


Convolution implementation of sliding windows





Bounding Box Predictions



* Label for training for each grid cell:

$$y = \begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix} \rightarrow \begin{bmatrix} 0 \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \\ ? \end{bmatrix} \quad \begin{bmatrix} 1 \\ b_x \\ b_y \\ b_h \\ b_w \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

0.4, 0.3 } between 0 and 1

0.9, 0.5 } Could be > 1

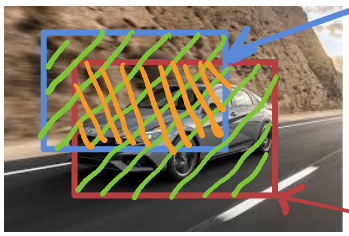
can be finer Grids.

→ X (e.g. 100 x 100 x 3) → CNN → 3 x 3 x 8 → y

input image — Target output

Intersection Over Union:

Intersection over Union (IoU) is used to evaluate object localization



$$IoU = \frac{\text{Size of Intersection}}{\text{Size of Union}} = \frac{\text{Size of } \blacksquare}{\text{Size of } \blacksquare}$$

Non-max Suppression algorithm

1) Discard all boxes with $p_c \leq 0.6$
2) While there are any remaining boxes:
   - Pick the box with the largest p_c output that as a prediction.
   - Discard any remaining box with IoU ≥ 0.5 with the box output in the previous step.

Conduct non-max suppression indepently for each object categories if the model detects multiple categories.

Anchor Boxes:

Allowing one grid to contain multiple objects.

Previously:

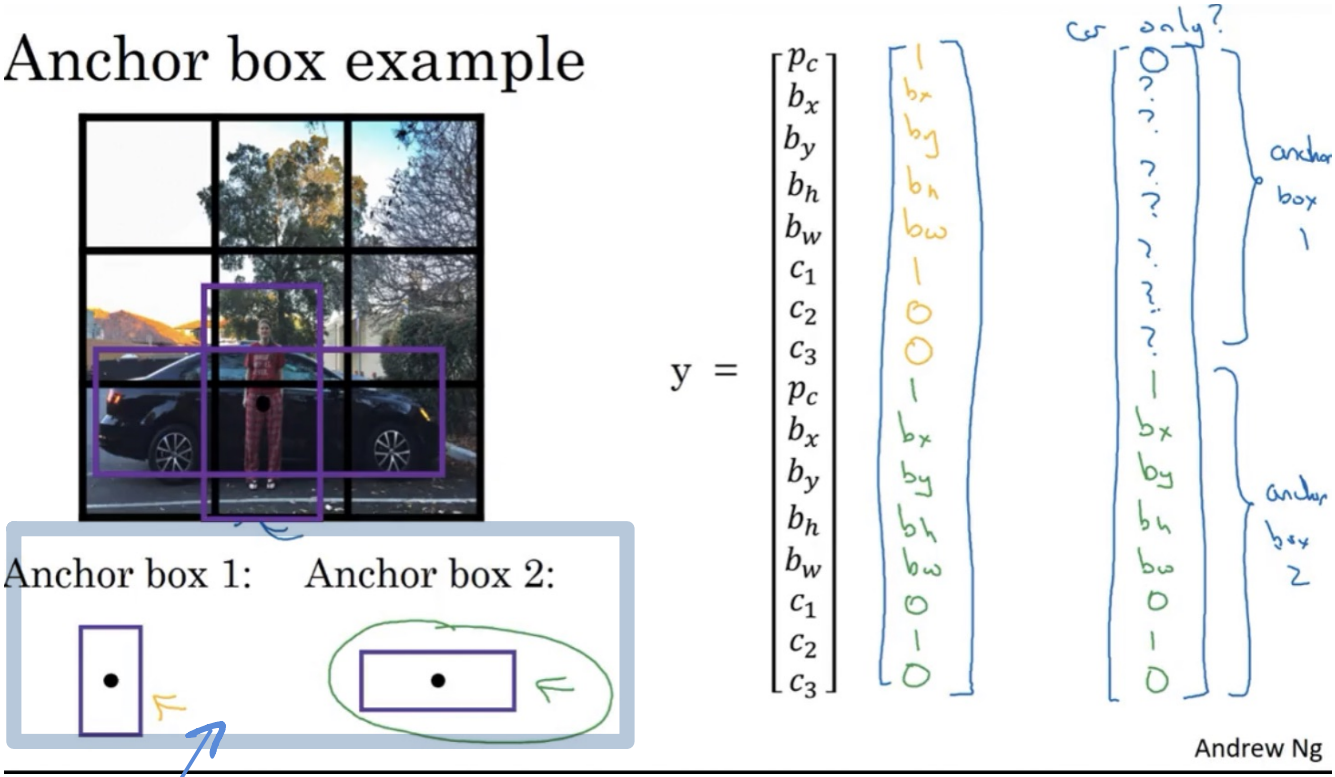
Each object in training image is assigned to grid cell that contains that object's midpoint

With 2 anchor boxes:

Each object in training image is assigned to grid cell that contains object's midpoint and anchor box for the grid cell with highest IoU

In this case: output y: $3 \times 3 \times \underbrace{16}_{2 \times 8}$

anchor boxes

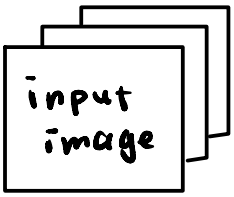


The number and shapes of Anchor boxes are defined before training, manually or algorithmically

YOLO Algorithm

Training :

Specify = Object categories (e.g. 1. pedestrian, 2. car, 3. motorcycle).
anchor boxes



→ ConvNet → $3 \times 3 \times 16$

Grid Size $2 \times 2 \times 8$ → $\begin{bmatrix} p_c \\ b_x \\ b_y \\ b_h \\ b_w \\ c_1 \\ c_2 \\ c_3 \end{bmatrix}$

anchor boxes

Making Predictions:

image → trained ConvNet → $3 \times 3 \times 2 \times 8$

Predictions

* Outputting the non-max suppressed outputs :

- For each grid cell, get 2 predicted bounding boxes
- Get rid of low probability predictions
- For each class, use non-max suppression to generate final predictions.

Region proposal = idea of semantic segmentation + CNN on segmentations to detect objects and boxes

R-CNN : Propose Regions. Classify proposed regions one at a time output label + bounding box

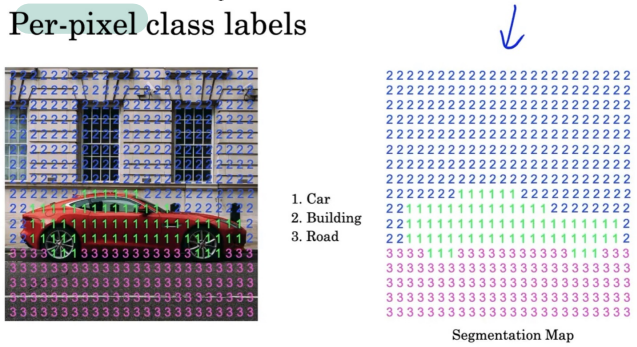
Fast R-CNN: Propose Regions, Use convolutional implementation of sliding window to classify all proposed regions.

Faster R-CNN: Use convolution network to propose regions

Semantic Segmentation with U-Net

Semantic Segmentation:

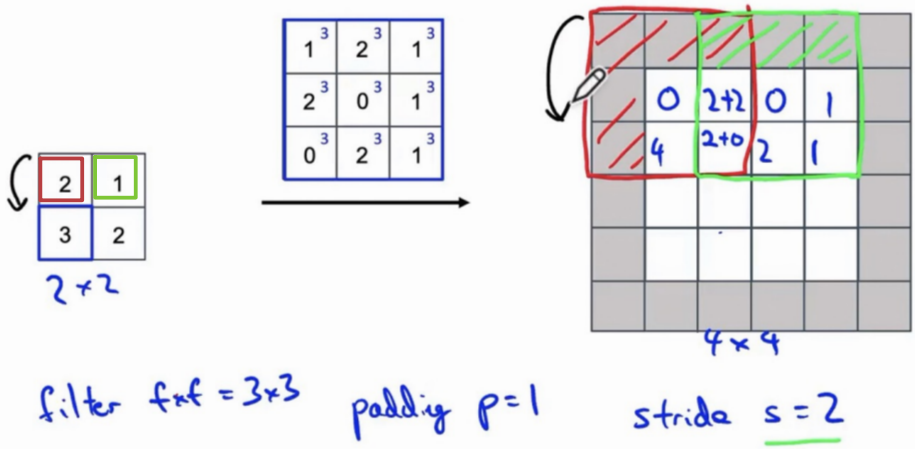
Per-pixel class labels



Andrew Ng

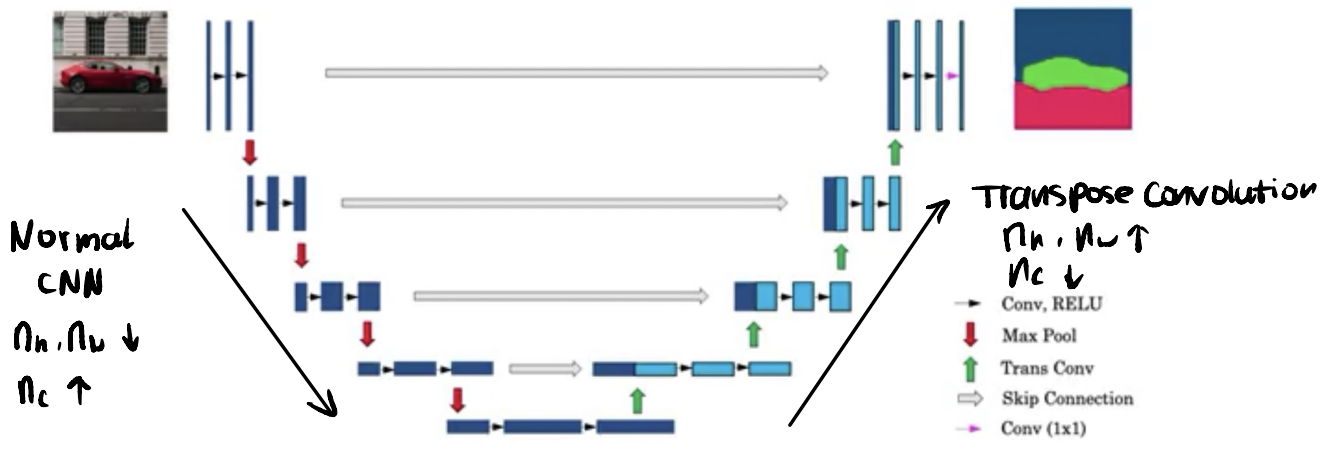
Challenge: Normal CNN would gradually decrease n_H and increase n_C. In order to output per-pixel class labels, we need a type of transformations to bring up n_H, n_W sizes back to the original dimensions.

→ Transpose Convolution



Andrew Ng

U-Net



[Ronneberger et al., 2015, U-Net: Convolutional Networks for Biomedical Image Segmentation]

Andrew Ng